From Sigmoid Power Control Algorithm to Hopfield-like Neural Networks: "SIR"-Balancing Sigmoid-Based Networks- Part II: Discrete Time

Zekeriya Uykan

Abstract

In the first part in [12], we present and analyse a Sigmoid-based ”Signal-to-Interference Ratio, (SIR)” balancing dynamic network, called Sgm”SIR”NN, which exhibits similar properties as traditional Hopfield NN does, in continuous time. In this second part, we present the corresponding network in discrete time: We show that in the proposed discrete-time network, called D-Sgm”SIR”NN, the defined error vector approaches to zero in a finite step in both synchronous and asynchronous work modes. Our investigations show that i) Establishing an analogy to the distributed (sigmoid) power control algorithm in [10] and [11] if the defined fictitious ”SIR” is equal to 1 at the converged eqiulibrium point, then it is one of the prototype vectors. ii) The D-Sgm”SIR”NN exhibits similar features as discrete-time Hopfield NN does. iii) Establishing an analogy to the traditional 1-bit fixed-step power control algorithm, the corresponding ”1-bit” network, called Sign”SIR”NN network, is also presented.

Index Terms

Discrete-time Hopfield Network, distributed sigmoid power control algorithm.

I. Introduction

This paper is a continuation of the study in [12] where a continuous-time ”Signal-to-Interference Ratio, (SIR)”-balancing neural network is presented which includes Hofield Network and sigmoid-

Z. Uykan is with Helsinki University of Technology, Control Engineering Laboratory, FI-02015 HUT, Finland. E-mail: zekeriya.uykan@hut.fi. The author is a visiting scientist at Harvard University Broadband Comm Lab., Cambridge, MA, and this work has been performed during his stay at Harvard University.

based power control algorithm algorithm of [10] and [11] as special cases, both of whose scopes of interest, motivations and settings are completely different. In this paper, we examine the discrete-time counterpart of [12], and propose two discrete-time sigmoid-basis SIR-balancing networks which exhibit similar features which generaly are attributed to recurrent neural networks like discrete-time Hopfield Networks.

Hopfield Neural Networks has been an important focus of research area since early 1980s whose applications vary from combinatorial optimization (e.g. [2], [3] among many others) including traveling salesman problem (e.g. [4], [16] among others) to image restoration (e.g. [5]), from various control engineering optimization problems including in robotics (e.g. [8] among others) to associative memory systems (e.g. [7] among others), etc. For a tutorial and further references about Hopfield NN, see e.g. [13] and [9].

In the first part in [12], we present a Sigmoid-based ”Signal-to-Interference Ratio (SIR)” balancing dynamic network, called Sgm”SIR”NN, which exhibits similar properties as traditional Hopfield NN does, is presented and analysed in continuous time. In this second part, we present the corresponding networks in discrete time. Our investigations show that i) Establishing an analogy to the distributed (sigmoid) power control algorithm in [10] and [11], if the defined fictitious ”SIR” is equal to 1 at the converged eqiulibrium point, then it is one of the prototype vectors. ii) The D-Sgm”SIR”NN exhibits similar features as discrete-time Hopfield NN does. iii) Establishing an analogy to the traditional 1-bit fixed-step power control algorithm, the corresponding ”1-bit” network, called Sign”SIR”NN network, is also presented.

The paper is organized as follows: The proposed D-Sgm”SIR”NN and its 1-bit version network is presented and their stability features are analysed in section II. Simulation results are presented in Section III followed by Concluding Remarks in Section IV.

II. “SIR”-Balancing Sigmoid-Based Networks in Discrete Time

We start with the standard definition of Signal-to-Interferende+Noise-Ratio (SIR) in a cellular radio system, in which N mobiles share the same channel (e.g. [17], [18]).

$$\gamma_i = \frac{g_{ii} p_i}{\nu_i + \sum_{j=1, j \neq i}^{N} g_{ij} p_j}, \quad i = 1, \ldots, N \tag{1}$$

where p_i is the transmission power of mobile i, g_{ij} is the link gain from mobile j to base i involving path loss, shadowing, multi-path fading (as well as the spreading/processing gain in case of CDMA transmission [6], etc), and ν_i is the receiver noise at base station i.

Because, in power control, the positive transmit power can not be arbitarily small and large in practice, we write the eq.(1) with the minimum and maximum power constraints as follows:

$$\bar{\gamma}_i = \frac{g_{ii} \max\{p_{min}, \min\{p_{max}, p_i\}\}}{\nu_i + \sum_{j=1, j\neq i}^{N} g_{ij} \max\{p_{min}, \min\{p_{max}, p_j\}\}}, \quad i = 1, \ldots, N \tag{2}$$

where p_{min} and p_{max} is the minimum and maximum transmit powers. The SIR model in (2) can be further written in a more generalized equation as follows using neural networks termonilogy

$$\bar{\gamma}_i = \frac{g_{ii} y(p_i)}{\nu_i + \sum_{j=1, j\neq i}^{N} g_{ij} y(p_j)}, \quad i = 1, \ldots, N \tag{3}$$

where $y(\cdot)$ represents the modeling of lower and upper bounding the transmit power and of any other effects e.g. power amplifier, etc. For example, $y(p_i) = \max\{p_{min}, \min\{p_{max}, p_i\}\}$ or corresponding piecewise linear function $y(p_i) = |p_i + p_{max}| - |p_i - p_{max}|$ yields eq.(2).

By relaxing the positivity conditions in the power control problem in (3) and using sigmoid as the bounding function to the states in the denominator, and a different function in the nominator, the following fictitious "SIR" is defined in [12]:

$$\bar{\theta}_i = \frac{a_{ii} f_3(x_i)}{b_i + \sum_{j=1, j\neq i}^{N} w_{ij} f_2(x_j)}, \quad i = 1, \ldots, N \tag{4}$$

where θ_i is the defined fictitious "SIR", x_i is the state of the i'th neuron, a_{ii} is the feedback coefficient from its state to its input layer, w_{ij} is the weight from the output of the j'th neuron to the input of the j'th neuron, and $f_2(\cdot)$ represents the sigmoid function, and $f_3(\cdot)$ represents the function used for self-state-feedback. Sigmoid function is defined as $f_2(e_i) = 1 - \frac{1}{1+exp(-\sigma_1 e_1)}$, where $\sigma_1 > 0$ is called slope of $f_2(\cdot)$, which is equal to its derivative with respect to its argument at the origin 0.

It's shown in [12] that choosing $f_3(\cdot)$ as a unity function in (4), i.e., $f_2(x_i) = x_i$, yields a network, called Sgm"SIR"NN which exhibits similar features as Hopfield NN does. So, following fictitious "SIR" is defined

$$\frac{\bar{\theta}_i}{\theta_i^{tgt}} = \frac{a_{ii}x_i}{b_i + \sum_{j=1,j\neq i}^{N} w_{ij} f_2(x_j)}, \quad i = 1, \ldots, N \tag{5}$$

which is shown to satisfy the equilibrium points (prototype vectors) of the following dynamic network with $\theta_i^{tgt} = 1$, called Sgm"SIR"NN in [12]:

$$\dot{\mathbf{x}} = \mathbf{f}_1\Big(- \mathbf{A}\mathbf{x} + \mathbf{W}\mathbf{f_2}(\mathbf{x}) + \mathbf{b}\Big) \tag{6}$$

where $\dot{\mathbf{x}}$ represents the derivative of $\mathbf{x}$ with respect to time and

$$\mathbf{A} = \begin{bmatrix} a_{11} & 0 & \ldots & 0 \\ 0 & a_{22} & \ldots & 0 \\ \vdots & & \ddots & 0 \\ 0 & 0 & \ldots & a_{NN} \end{bmatrix}, \qquad \mathbf{W} = \begin{bmatrix} 0 & w_{12} & \ldots & w_{1N} \\ w_{21} & 0 & \ldots & w_{2N} \\ \vdots & & \ddots & \vdots \\ w_{N1} & w_{N2} & \ldots & 0 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix} \tag{7}$$

In eq.(7), $\mathbf{A}$ shows the self-state-feedback matrix with $a_{jj} > 0$, $\mathbf{W}$ with zero diagonal shows the connection weight matrix from outputs to other neuron's inputs, and $\mathbf{b}$ is a threshold vector.

It's shown in [12] that the network in (6) exhibits similar features as continuous Hopfield Network does. In this paper, we examine its discrete-time version.

From the fictitious CIR definition in eq.(5), let's define the following error signal

$$e_i = -a_{ii}x_i + I_i, \qquad \text{where} \quad I_i = b_i \sum_{j=1,j\neq i}^{N} w_{ij} f_2(x_j), \quad i = 1, \ldots, N \tag{8}$$

Writing (8) in matrix form gives

$$\mathbf{e} = -\mathbf{A}\mathbf{x} + \mathbf{W}\mathbf{f}_2(\mathbf{x}) + \mathbf{b} \tag{9}$$

which is equal to the argument of the $\mathbf{f}_1(\cdot)$ in the network Sgm"SIR"NN in eq.(6).

From eq. (6) and (9), $\dot{\mathbf{x}} = \mathbf{e}$. If $e_i = 0$ given that $x_i \neq 0$ and $I_i \neq 0$, then, from eq.(5) and (8), $\hat{\theta}_i = \theta_i^{tgt} = 1$.

The prototype vectors are defined as those $\mathbf{x}$'s which make $\theta_i = \theta_i^{tgt} = 1, \quad i = 1, \ldots, N$ given that $x_i \neq 0$ and $I_i \neq 0$. So, from (4) and (5), the prototype vectors make the error signal zero, i.e., $e_i = 0, \quad i = 1, \ldots, N$.

A. Discrete Sgm"SIR"NN Network

In this section, we present a Sigmoid based "SIR"-balancing network which exhibits similar features as discrete Hopfield NN does.

Discretizing the differential equation (6) by the Euler method gives

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha \mathbf{f}_1\Big(-\mathbf{A}\mathbf{x}^k + \mathbf{W}\mathbf{f}(\mathbf{x^k}) + \mathbf{b}\Big) \tag{10}$$

where $\mathbf{A}, \mathbf{W}$ and $\mathbf{b}$ are defined as in eq.(7), and k represents the iteration step.

From eq.(10) and (7),

$$x_j^{k+1} = x_j^k + \alpha^k f_1\Big(-a_{jj}x_j^k + b_j + \sum_{i=1, i\neq j}^{N} w_{ij} f_2(x_i^k)\Big) \qquad j = 1, \ldots, N \tag{11}$$

where α^k is the step size at time k.

We will call the network in eq.(11) as D-Sgm"SIR"NN (Discrete Sigmoid "SIR"-balancing neural network).

The performance index is defined as l_1-norm of the error vector in (9) as follows

$$V(k) = ||\mathbf{e}(k)||_1 = \sum_{i}^{N} |e_i(k)| \tag{12}$$

$$= \sum_{i}^{N} |-a_{ii}x_i + I_i| \quad \text{where} \quad I_i = b_i + \sum_{j=1, j\neq i}^{N} w_{ij} f_2(x_j) \tag{13}$$

In what follows, we examine the evolution of the the energy function in (12) in synchronous and asynchronous work modes. Synchronous mode means that at every iteration step, at most only one state is updated, whereas asynchronous mode refers to the fact that all the states are updated at every iteration step according to eq.(11).

Proposition 1:

In asynchronous mode, in the D-Sgm"SIR"NN in eq.(11) with a symmetric matrix $\mathbf{W}$, the l_1-norm of the error vector in eq.(12) decreases at every step for a nonzero error vector, i.e., the error vector goes to zero for any α^k such that

$$|e_j^k| > |a_{jj}\alpha^k f_1(e_j^k)| \tag{14}$$

if

$$|a_{jj}| \geq k_2 \sum_{i=1,(i\neq j)}^{N} |w_{ij}| \tag{15}$$

where $k_2 = 0.5\sigma$ is the the global Lipschitz constant of $f_2(\cdot)$ as shown the in Appendix A.

Proof:

In asynchronous mode, only one state is updated at an iteration time. Let j shows the state which is updated at time k whose error signal is different than zero, i.e., $e_j = -a_{jj}x_j + I_j \neq 0$, where $I_j = b_j \sum_{i=1,i\neq j}^{N} w_{ji} f_2(x_i)$, as defined in eq.(8).

Using eq.(9), we get

$$\mathbf{e}^{k+1} - \mathbf{e}^{k} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ -a_{11}(x_j^{k+1} - x_j^k) \\ \vdots \\ 0 \end{bmatrix} + \begin{bmatrix} w_{1j} \\ w_{2j} \\ \vdots \\ 0 \\ \vdots \\ w_{Nj} \end{bmatrix} (f_2(x_j^{k+1}) - f_2(x_j^k)) \tag{16}$$

Using the error signal definition of eq.(8) in eq.(11) gives

$$x_j^{k+1} - x_j^k = \alpha f_1(e_j^k) \tag{17}$$

So, the error signal for state j is obtained using eq.(16) and (17) as follows

$$e_j^{k+1} - e_j^k = -a_{jj}(x_j^{k+1} - x_j^k) \tag{18}$$

$$= -a_{jj}\alpha f_1(e_j^k) \tag{19}$$

From eq.(18) and (19), if α is chosen to satisfy $|e_j^k| > |a_{jj}\alpha f_1(e_j^k)|$, then

$$|e_j^{k+1}| < |e_j^k|, \quad \text{for} |e_i^k| \neq 0 \tag{20}$$

Since sigmoid function $f_1(\cdot)$ is an increasing odd function and $f_1(e_j) = 0$ if and only if $e_j = 0$, then it's seen that there α can easily be chosen small enough to satisfy $|e_j^k| > \alpha a_{jj}|f_1(e_j^k)|$ according to the parameter a_{jj} and slope of sigmoid function $f_1(\cdot)$.

Above, we examined only the state j and its error signal $e_j(k)$. In what follows, we examine the evolution of the norm of the complete error vector $\mathbf{e}^{k+1}$ in eq.(16). From the point of view of the l_1 norm of the $\mathbf{e}^{k+1}$, the worst case is that while $|e_j^k|$ decreases, all other elements $|e_i^k|, \quad i \neq j$, increases. So, using eq.(16), (18) and (20), we obtain that: If

$$|-a_{jj}(x_j^{k+1}-x_j^k)| \geq |f_2(x_j^{k+1})-f_2(x_j^k)| \sum_{i=1,(i\neq j)}^{N} |w_{ij}| \tag{21}$$

then

$$||\mathbf{e}(k+1)||_1 \begin{cases} < ||\mathbf{e}(k)||_1 & \text{if} ||\mathbf{e}(k)||_1 \neq \mathbf{0} \\ = 0 & \text{if} ||\mathbf{e}(k)||_1 = \mathbf{0} \end{cases} \tag{22}$$

The sigmoid function $f_2(\cdot)$ is a Lipschitz continuous function as shown in Appendix A. So,

$$k_2|x_j^{k+1}-x_j^k| \geq |f_2(x_j^{k+1})-f_2(x_j^k)| \tag{23}$$

where $k_2 = 0.5\sigma$ is $f_2(\cdot)$'s global Lipschitz constant as shown in Appendix A.

From eq.(21) and (23), choosing $|a_{jj}| > k_2|\sum_{i=1,(i\neq j)}^{N}|w_{ij}|$ yields eq.(21), which implies eq.(22). This completes the proof.

■

Proposition 2:

In asynchronous mode, choosing the slope of $f_2(\cdot)$ relatively small as compared to $f_1(\cdot)$ and choosing $a_{jj} > 0$ and α satisfying (14), the D-Sgm"SIR"NN in eq.(11) with a symmetric matrix $\mathbf{W}$ is stable and there exists a finite step number T_d such that the l_1-norm of the error vector in eq.(12) goes to zero as its steady state. If $\bar{\theta}_i = \theta_i^{tgt1} = 1$ at the converged point, then it corresponds to a prototype vector as defined above.

Proof:

Since it's asynchronous mode, eqs.(16)-(20) holds where $a_{jj} > 0$. So, if α^k at time k is chosen to satisfy $|e_j^k| > |a_{jj}\alpha^k f_1(e_j^k)|$ as in (14), then

$$|e_j(k+1)| < |e_j(k)|, \quad \text{for} \quad |e_i(k)| \neq 0 \tag{24}$$

Note that it's straighforward to choose a sufficiently small α^k to satisfy (14) according to a_{jj} and the slope σ of sigmoid $f_1(\cdot)$.

Using eq.(16), (18) and (24), it's seen for $e_j^k \neq 0$ that: If

$$\begin{aligned} |-a_{jj}(x_j^{k+1} - x_j^k)| &= |-a_{jj}\alpha f_1(e_j^k)| & (25) \\ &> |f_2(x_j^{k+1}) - f_2(x_j^k)| \sum_{i=1,(i\neq j)}^{N} |w_{ij}| & (26) \end{aligned}$$

then

$$||\mathbf{e}(k+1)||_1 < ||\mathbf{e}(k)||_1 \tag{27}$$

We observe from eq.(18), (25), (26) and (27) that:

1) If the $x_i^k, \quad i = 1, \ldots, N$, approach to either of the saturation regimes of its sigmoid function $f_2(\cdot)$, then

$$|f_2(x_j^{k+1}) - f_2(x_j^k)| \sum_{i=1,(i\neq j)}^{N} |w_{ij}| \approx 0, \qquad j = 1, \ldots, N \tag{28}$$

since $|f_2(x_j^{k+1}) - f_2(x_j^k)| \approx 0, \qquad i = 1, \ldots, N$. That makes eq.(25) and (26) hold. Therefore, the norm of the error vector in eq.(12) does not go to infinity, and is finite for any $\mathbf{x}$.

2) $\mathbf{x}(k+1) = \mathbf{x}(k)$ if and only if $\mathbf{e}(k) = \mathbf{0}$, i.e.,

$$x_j^{k+1} = x_j^k \quad \text{if and only if} \quad f_1(e_j^k) = 0, \qquad j = 1, \ldots, N \tag{29}$$

3) Examining the eq.(17), (18) and (19) taking the observations 1 and 2 into account, we conclude that any of the $x_j^k, \quad j = 1, \ldots, N$, does not go to infinity, and is finite for any k. So, the D-Sgm"SIR"NN in eq.(11) with a symmetric matrix $\mathbf{W}$ is stable for the assumptions in proposition 2. Because there is a finite number of in-saturation states, (i.e. the number of all possible in-saturation state combinations is finite), which is equal to 2^N, there exists a finite step number, say T_d, such that $\mathbf{e}(t) = 0$ for any $t \geq T_d$.

From eq.(5), if $\bar{\theta}_i = \theta_i^{tgt1} = 1$ at the converged point, then it corresponds to a prototype vector as defined in previous section, which completes the proof.

■

In what follows, we examine the evolution $\bar{\theta}_i^k$. From eq.(5), by choosing $\theta_j^{tgt} = 1$, let's define the following error signal at time k

$$\xi_j^k = -\theta_j^k + \theta_j^{tgt} = -\theta_j^k + 1, \qquad j = 1, \ldots, N \tag{30}$$

Lemma 1:

In asynchronous mode, in the D-Sgm"SIR"NN in eq.(11) with a sufficiently small α^k and with a symmetric matrix $\mathbf{W}$, the ξ_k is getting closer to $\theta_j^{tgt} = 1$ at those iteration steps k where $I_j^k \neq 0$, i.e., $|\xi_j(k+1)| < |\xi_j(k)|$, where index j shows the state being updated at iteration k.

Proof:

Let j shows the state which is updated at time k. The fictitious "SIR" is defined by eq.(5) for nonzero I_j^k as follows

$$\bar{\theta}_j^k = \frac{a_{ii} x_j^k}{I_j^k}, \quad \text{where} \quad I_j^k = b_j + \sum_{i=1, i \neq j}^{N} w_{ji} f(x_j^k) \tag{31}$$

Let's define the following error signal in $\bar{\theta}_j$, named ξ_j, as follows

$$\xi_j^k = -\bar{\theta}_j^k + 1 = \frac{-a_{ii} x_j^k + I_j^k}{I_j^k} \tag{32}$$

In asynchronous mode, from eq.(31), $I_m^k = I_m^{k+1}$. Using this observation and eq.(32)

$$\xi_j^{k+1} - \xi_j^k = \frac{-a_{ii}(x_j^{k+1} - x_j^k)}{I_j^k} \tag{33}$$

From (11) and (33),

$$\xi_j^{k+1} - \xi_j^k = \frac{-a_{ii} \alpha f_1(e_j^k)}{I_j^k} \tag{34}$$

Provided that $I_j^k \neq 0$, we write from eq.(8) and (32),

$$e_j^k = I_j^k \xi_j^k \tag{35}$$

Writing eq.(35) in (34) gives

$$\xi_j^{k+1} - \xi_j^k = \frac{-a_{ii}\alpha f_1(I_j^k \xi_j^k)}{I_j^k} \tag{36}$$

From (36), since sigmoid function $f_1(\cdot)$ is an odd function, and $a_{ii} > 0$ and $\alpha > 0$,

$$\xi_j^{k+1} = \xi_j^k - \beta sign(\xi_j^k) \quad \text{where} \quad \beta = |\frac{-a_{ii}(\alpha f_1(I_j^k \xi_j^k))}{I_j^k}| \tag{37}$$

As seen from eq.(37), for a nonzero ξ_j^k, choosing a sufficiently small α satisfying $|\xi_j^k| > \beta$ assures that

$$|\xi_j^{k+1}| < |\xi_j^k| \quad \text{if} \quad I_m^k \neq 0 \tag{38}$$

which completes the proof.

■

Proposition 3:

In asynchronous mode, provided that the D-Sgm"SIR"NN in eq.(11) with a symmetric matrix $\mathbf{W}$ converges to one of the prototype vectors according to proposition 1 and 2,

$$\bar{\theta}_i^k = 1, \quad j = 1, \ldots, N \tag{39}$$

if and only if $I_i^k \neq 0, \quad j = 1, \ldots, N$ for the converged prototype vector.

Proof:

Proposition 1 and 2 shows that the norm of e_i^k decreases and approaches to zero in a finite step number and lemma 1 shows the norm of ξ_i^k also decreases if $I_i^k \neq 0$. From eq.(35), $e_i^k = I_i^k \xi_i^k$: As the e_i^k approaches to zero, then ξ_i^k also approaches to zero, given that $I_i^k \neq 0$. This is sketched by the following equation

$$\xi_i^k = 0, \quad (i.e., \bar{\theta}_i^k = 1) \qquad \text{if} \quad e_i^k = 0, \quad \text{and} \quad I_i \neq 0 \tag{40}$$

(41)

On the other hand, if $\xi_i^k = 0$ at the converged fixed point, then $e_i^k = 0$ because from eq.(35), $e_i^k = I_i^k \xi_i^k$

$$e_i^k = 0 \qquad \text{if} \quad \xi_i^k = 0 \tag{42}$$

provided that $I_i \neq 0, \quad j = 1, \ldots, N$, which completes the proof.

■

Proposition 4:

The results in proposition 1 and 2 for asynchronous mode hold also for synchronous mode.

Proof:

In asynchronous mode, from eq.(9)

$$\mathbf{e}^{k+1} - \mathbf{e}^k = \sum_{i=1}^{N} \Bigg(\begin{bmatrix} 0 \\ 0 \\ \vdots \\ -a_{11}(x_i^{k+1} - x_i^k) \\ \vdots \\ 0 \end{bmatrix} + \begin{bmatrix} w_{1i} \\ w_{2i} \\ \vdots \\ 0 \\ \vdots \\ w_{Ni} \end{bmatrix} (f_2(x_i^{k+1}) - f_2(x_i^k)) \Bigg) \tag{43}$$

Using (8) in eq.(43) and writing elementwise gives

$$e_i^{k+1} = e_i^k - a_{ii}\alpha f_1(e_i^k) + \sum_{j=1,(j \neq i)}^{N} w_{ij}(f_2(x_j^{k+1} - f_2(x_j^k)), \quad i = 1, \ldots, N \tag{44}$$

From eq.(43) and (44), we obtain

$$|-a_{ii}(x_i^{k+1} - x_i^k)| = |-a_{ii}\alpha f_1(e_i^k)| \tag{45}$$

$$> |f_2(x_i^{k+1}) - f_2(x_i^k)| \sum_{j=1,(j \neq i)}^{N} |w_{ji}| \quad i = 1, \ldots, N \tag{46}$$

which is equal to (21) in proposition 1 and (25) in proposition 2. Continueing the the steps of the analysis in proposition 1 and proposition 2 yield the results in proposition 1 and proposition 2 respectively.

■

B. Fixed-Step Discrete“SIR”NN Network (FS”SIR”NN)

In this subsection, establishing an analogy to the traditional fixed step 1-bit increase/decrease power control algorithm e.g. [22], [23], we propose the following network by replacing the $f_1(\cdot)$ in eq.(11) by sign function as shown in the following

$$x_j^{k+1} = x_j^k + \Delta sign\Big(- a_{jj}x_j^k + b_j + \sum_{i=1, i\neq j}^{N} w_{ji} f_2(x_j^k)\Big) \qquad j = 1, \ldots, N \tag{47}$$

where $f_2(\cdot)$ represents the sigmoid function. We call the network in eq.(47) as Fixed-Step “SIR” Neural Networks (FS”SIR”NN).

Corollary 1:

In the FS”SIR”NN in eq.(47) with a symmetric matrix $\mathbf{W}$, the l_1-norm of the error vector in eq.(12) converges to the interval $[-a_{ii}\alpha, +a_{ii}\alpha]$ while the x_i^k converges to the interval $[\alpha, +\alpha]$ within a finite step number in asynchoronous mode if

$$|a_{jj}| \geq k_2 \sum_{i=1,(i\neq j)}^{N} |w_{ij}| \tag{48}$$

where $k_2 = 0.5\sigma$ is the the global Lipschitz constant of $f_2(\cdot)$ as shown the in Appendix A.

Proof:

We’re going to obtain the results by writing $f_1(\cdot) = sign(\cdot)$ in the proof of proposition 1 in section II-A above. This would correspond to a sigmoid function $f_1(\cdot)$ whose slope is infinity in proposition 1.

Let j show the state which is updated at time k. Following the steps in eq.(9) and (16) and writing $f_1(\cdot) = sign(\cdot)$ in eq.(17) gives

$$x_j^{k+1} - x_j^k = \alpha sign(e_j^k) \tag{49}$$

So, the error signal for state j is obtained using eq.(16) and 49 as follows

$$\begin{aligned} e_j^{k+1} - e_j^k &= -a_{jj}(x_j^{k+1} - x_j^k) && (50) \\ &= -a_{jj}\alpha sign(e_j^k) && (51) \end{aligned}$$

From eq.(50) and (51),

$$|e_j^{k+1}| \begin{cases} < |e_j^k| & \text{if} |e_j^k| > |\alpha a_{jj}| \\ < |\alpha a_{jj}| & \text{otherwise} \end{cases} \tag{52}$$

Above, we examined only the state j and its error signal $e_j(k)$. In what follows, we examine the evolution of the norm of the complete error vector $\mathbf{e}^{k+1}$ in eq.(16). From the point of view of the l_1 norm of the $\mathbf{e}^{k+1}$, the worst case is that while $|e_j^k|$ decreases, all other elements $|e_i^k|, \quad i \neq j$, increases. So, using eq.(16), eq.(50-(52), we obtain that: If

$$| - a_{jj}(x_j^{k+1} - x_j^k)| \geq |f_2(x_j^{k+1}) - f_2(x_j^k)| \sum_{i=1,(i \neq j)}^{N} |w_{ij}| \tag{53}$$

then

$$||\mathbf{e}(k+1)||_1 \begin{cases} < ||\mathbf{e}(k)||_1 & \text{if} ||\mathbf{e}(k)||_1 > \Delta \sum_{i=1,(i\neq j)}^{N} |a_{ii}| \\ < \Delta \sum_{i=1,(i\neq j)}^{N} |a_{ii}| & \text{otherwise} \end{cases} \tag{54}$$

The sigmoid function $f_2(\cdot)$ is a Lipschitz continuous function as shown in Appendix A, $k_2|x_j^{k+1} - x_j^k| \geq |f_2(x_j^{k+1}) - f_2(x_j^k)|$, where $k_2 = 0.5\sigma$ is $f_2(\cdot)$. From eq.(53) and the Lipschitz inequality, choosing $|a_{jj}| > k_2| \sum_{i=1,(i\neq j)}^{N} |w_{ij}|$ satisfies eq.(53), which implies eq.(54). This completes the proof.

■

Corollary 2:

In asynchronous mode, choosing $f_1(e_i) = sign(e_i)$, $a_{jj} > 0$ and $\alpha > 0$, the FS”SIR”NN in eq.(47) with a symmetric matrix $\mathbf{W}$ is stable and there exists a finite step number T_d such that the l_1-norm of the error vector in eq.(12) converges to the interval $[-a_{ii}\alpha, +a_{ii}\alpha]$ while the x_i^k converges to the interval $[\alpha, +\alpha]$ within a finite step number.

Proof:

Writing $f_1(\cdot) = sign(\cdot)$ in proposition 2 in section II-A and following the steps and the observations therein gives that the FS”SIR”NN in eq.(47) is stable and there exists a finite step number T_d such that the l_1-norm of the error vector in eq.(12) converges to the interval $[-a_{ii}\alpha, +a_{ii}\alpha]$ while the x_i^k converges to the interval $[\alpha, +\alpha]$.

■

Corollary 3:

The results in corollary 1 and 2 for asynchronous mode hold also for synchronous mode.

Proof:

Writing $f_1(\cdot) = sign(\cdot)$ in proposition 1 and 2 in section II-A and following the the steps and observations of the analysis as in proposition 4 in section II-A for synchronous mode yields the results in corollary 1 and 2 respectively above.

■

It's known form literature that the performance of Hopfield network may highly depend on the parameter setting of the weight matrix (eg. [15]). There are various ways for determining the weight matrix of the Hopfield Networks: Gradient-descent supervised learning (e.g. [9]), solving linear inequalities (e.g. [20], [21] among others), Hebb learning rule [19], [14] etc. How to design CINR-SgmNN is out of the scope of this paper. The methods used for traditional Hopfied NN can also be used for the proposed networks D-Sgm“CIR”NN and FS“CIR”NN.

III. Simulation Results

We take the same examples as in [12] for comparison reasons and for the sake of brevity and easy reproduction of the simulation results. In [12], the performances of continuous-time networks, Sgm”SIR”NN and Hopfield networks, are examined. In this paper, their discrete-time versions are examined. We apply the same Hebb-based (outer-products-based) design procedure ([19]) in [12], which is presented in Appendix B in case of orthogonal prototype vectors.

In this section, we present two examples, one with 8 neurons and one with 16 neurons. The weight matrices are designed by the outer products-based design in Appendix B.

As in [12], traditional Hopfield network is used a reference network. The discrete Hopfield Network [1] is

$$\mathbf{x}^{k+1} = sign\Big(\mathbf{W}\mathbf{x}^k\Big) \tag{55}$$

where $\mathbf{W}$ is the weight matrix and $\mathbf{x}^k$ is the state at time k, and at most one state is updated.

Example 1:

In this example, there are 8 neurons. The desired prototype vectors are

$$\mathbf{D} = \begin{bmatrix} 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} \tag{56}$$

The weight matrices $\mathbf{A}$ and $\mathbf{W}$, and the threshold vector $\mathbf{b}$ are obtained as follows by using the outer-products-based design presented in Appendix B and the slopes of sigmoid functions $f_1(\cdot)$ and $f_2(\cdot)$ are set to $\sigma_1 = 10$ and $\sigma_2 = 2$ respectively, and $\rho = 0$, $\alpha = 0.1$ and $\Delta = 0.1$.

$$\mathbf{A} = 3\mathbf{I}, \qquad \mathbf{W} = \begin{bmatrix} 0 & 1 & 1 & -1 & 1 & -1 & -1 & -3 \\ 1 & 0 & -1 & 1 & -1 & 1 & -3 & -1 \\ 1 & -1 & 0 & 1 & -1 & -3 & 1 & -1 \\ -1 & 1 & 1 & 0 & -3 & -1 & -1 & 1 \\ 1 & -1 & -1 & -3 & 0 & 1 & 1 & -1 \\ -1 & 1 & -3 & -1 & 1 & 0 & -1 & 1 \\ -1 & -3 & 1 & -1 & 1 & -1 & 0 & 1 \\ -3 & -1 & -1 & 1 & -1 & 1 & 1 & 0 \end{bmatrix}, \qquad \mathbf{b} = \mathbf{0} \tag{57}$$

The Figure 1 shows the percentages of correctly recovered desired patterns for all possible initial conditions $\mathbf{x}^k \in (-1, +1)^N$, in the proposed networks D-Sgm"SIR"NN and FS-Sgm"SIR"NN as compared to traditional discrete Hopfield network.

Let m_d show the number of prototype vectors and $C(N, K)$, (such that $N \geq K \geq 0$), represent the combination N, K, which is equal to $C(N, K) = \frac{N!}{(N-K)!K!}$, where ! shows factorial. In our simulation, the prototype vectors are from $(-1, 1)^N$ as seen above. For initial conditions, we alter the sign of K states where K=0, 1, 2, 3 and 4, which means the initial condition is within K-Hamming distance from the corresponding prototype vector. So, the total number of different possible combinations for the initial conditions for this example is 24, 84 and 168 for 1, 2 and 3-Hamming distance cases respectively, which could be calculated by $m_d \times C(8, K)$, where $m_d = 3$ and $K =$ 1, 2 and 3.

As seen from Figure 1 the performance of the proposed network D-Sgm"SIR"NN is remarkably better than that of traditinal discrete Hopfield NN for 1, 2 and 3 Hamming distance cases. The FS"SIR"NN also considerably outperforms the Hopfield for 1 and 2 Hamming distance cases while Hopfield NN outperforms FS"SIR"NN at 3 Hamming distance case.

Example 2:

The desired prototype vectors are

$$\mathbf{D} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} \quad (58)$$

The weight matrices $\mathbf{A}$ and $\mathbf{W}$ and threshold vector $\mathbf{b}$ is obtained as follows by using the outer products based design explained above.

$$\begin{aligned} \mathbf{A} &= 4\mathbf{I}, \\ \mathbf{W} &= \begin{bmatrix} 0 & 2 & 2 & 0 & 2 & 0 & 0 & -2 & 2 & 0 & 0 & -2 & 0 & -2 & -2 & -4 \\ 2 & 0 & 0 & 2 & 0 & 2 & -2 & 0 & 0 & 2 & -2 & 0 & -2 & 0 & -4 & -2 \\ 2 & 0 & 0 & 2 & 0 & -2 & 2 & 0 & 0 & -2 & 2 & 0 & -2 & -4 & 0 & -2 \\ 0 & 2 & 2 & 0 & -2 & 0 & 0 & 2 & -2 & 0 & 0 & 2 & -4 & -2 & -2 & 0 \\ 2 & 0 & 0 & -2 & 0 & 2 & 2 & 0 & 0 & -2 & -2 & -4 & 2 & 0 & 0 & -2 \\ 0 & 2 & -2 & 0 & 2 & 0 & 0 & 2 & -2 & 0 & -4 & -2 & 0 & 2 & -2 & 0 \\ 0 & -2 & 2 & 0 & 2 & 0 & 0 & 2 & -2 & -4 & 0 & -2 & 0 & -2 & 2 & 0 \\ -2 & 0 & 0 & 2 & 0 & 2 & 2 & 0 & -4 & -2 & -2 & 0 & -2 & 0 & 0 & 2 \\ 2 & 0 & 0 & -2 & 0 & -2 & -2 & -4 & 0 & 2 & 2 & 0 & 2 & 0 & 0 & -2 \\ 0 & 2 & -2 & 0 & -2 & 0 & -4 & -2 & 2 & 0 & 0 & 2 & 0 & 2 & -2 & 0 \\ 0 & -2 & 2 & 0 & -2 & -4 & 0 & -2 & 2 & 0 & 0 & 2 & 0 & -2 & 2 & 0 \\ -2 & 0 & 0 & 2 & -4 & -2 & -2 & 0 & 0 & 2 & 2 & 0 & -2 & 0 & 0 & 2 \\ 0 & -2 & -2 & -4 & 2 & 0 & 0 & -2 & 2 & 0 & 0 & -2 & 0 & 2 & 2 & 0 \\ -2 & 0 & -4 & -2 & 0 & 2 & -2 & 0 & 0 & 2 & -2 & 0 & 2 & 0 & 0 & 2 \\ -2 & -4 & 0 & -2 & 0 & -2 & 2 & 0 & 0 & -2 & 2 & 0 & 2 & 0 & 0 & 2 \\ -4 & -2 & -2 & 0 & -2 & 0 & 0 & 2 & -2 & 0 & 0 & 2 & 0 & 2 & 2 & 0 \end{bmatrix}, \\ \mathbf{b} &= \mathbf{0} \end{aligned} \quad (59)$$

The Figure 2 shows percentage of correctly recovered desired patterns for all possible initial conditions $\mathbf{x}^k \in (-1, +1)^16$, in the proposed D-Sgm"SIR"NN and FS"SIR"NN as compared to

discrete Hopfield network.

The total number of different possible combinations for the initial conditions for this example is 64, 480 and 2240 and 7280 for 1, 2, 3 and 4-Hamming distance cases respectively, which could be calculated by $m_d \times C(16, K)$, where $m_d = 4$ and $K =$ 1, 2, 3 and 4.

As seen from Figure 2 the performance of the proposed networks D-Sgm"SIR"NN and FS"SIR"NN is the same as that of discrete Hopfield Network for 1-Hamming and 2-Hamming distance cases ($\%100$ for all networks). However, the D-Sgm"SIR"NN and FS"SIR"NN gives better performance than the discrete Hopfield network does for 3 and 4 Hamming distance cases.

IV. Concluding Remarks

This paper is continuation of the work in [12] where we present and analyse a Sigmoid-based "Signal-to-Interference Ratio, (SIR)" balancing dynamic network in continuous time. In this second part, we present the corresponding network in discrete time: We show that in the proposed discrete-time network, called D-Sgm"SIR"NN, the defined error vector approaches to zero in a finite step in both synchronous and asynchronous work modes. Our investigations show that i) Establishing an analogy to the distributed (sigmoid) power control algorithm in [10] and [11], if the defined fictitious "SIR" is equal to 1 at the converged eqiulibrium point, then it is one of the prototype vectors. ii) The D-Sgm"SIR"NN exhibits similar features as discrete-time Hopfield NN does. iii) Establishing an analogy to the traditional 1-bit fixed-step power control algorithm, the corresponding "1-bit" network, called Sign"SIR"NN network, is also presented. Computer simulations show the effectiveness of the proposed networks as compared to traditional discrete Hopfield Network.

Appendix A

In what follows, we will show the sigmoid function ($f_2(a) = 1 - \frac{2}{1+exp(-\sigma a)}, \quad \sigma > 0$) has the global Lipschitz constant $k = 0.5\sigma$.

Since $f(\cdot)$ is a differentiable function, we can apply the mean value theorem

$$f(a) - f(b) = (a-b)f^{'}(\mu a + (1-\mu)(b-a))$$
$$with \quad \mu \in [0, 1]$$

The derivative of $f(\cdot)$ is $f^{'}(a) = \frac{\sigma}{e^{\sigma a}(1+e^{\sigma a})^2}$ whose maximum is at the point $a = 0$, i.e., $|f^{'}(a)| \leq 0.5\sigma$. So we obtain the following inequality

$$|f(a) - f(b)| \leq k|a - b| \tag{60}$$

where $k = 0.5\sigma$ is the global Lipschitz constant of the sigmoid function.

APPENDIX B

Outer products based network design:

Let's assume that L desired orthogonal prototype vectors, $\{\mathbf{d}_s\}_{s=1}^{L}$, are chosen form $(-1, +1)^N$.

Step 1: Calculate the sum of outer products of the prototype vectors (Hebb Rule, [19])

$$\mathbf{Q} = \sum_{s=1}^{L} \mathbf{d}_s \mathbf{d}_s^T \tag{61}$$

Step 2: Determine the diagonal matrix $\mathbf{A}$ and $\mathbf{W}$ as follows:

$$a_{ij} = \begin{cases} q_{ii} + \rho & \text{if} \quad i = j, \\ 0 & \text{if} \quad i \neq j \end{cases} \qquad i, j = 1, \ldots, N \tag{62}$$

where ρ is a real number and

$$w_{ij} = \begin{cases} 0 & \text{if} \quad i = j, \\ q_{ij} & \text{if} \quad i \neq j \end{cases} \qquad i, j = 1, \ldots, N \tag{63}$$

where q_{ij} shows the entries of matrix $\mathbf{Q}$, N is the dimension of the vector $\mathbf{x}$ and L is the number of the prototype vectors ($N > L > 0$). In eq.(62), $q_{ii} = L$ from (61) since $\{\mathbf{d}_s\}$ is from $(-1, +1)^N$ and ρ is a real number. However, from the analysis in section II-A and II-B, it can be seen that the proposed networks D-Sgm"SIR"NN and FS"SIR"NN contain the prototype vectors as their equilibrium points for a relatively large interval of ρ.

Another choice of ρ in (62) is $\rho = N - 2L$ which yields $a_{ii} = N - L$. In what follows we show that this choice also assures that $\{\mathbf{d}_j\}_{j=1}^{L}$ are the equilibrium points of the networks.

From (61)-(63)

$$[-\mathbf{A} + \mathbf{W}] = -(N - L)\mathbf{I} + \sum_{s=1}^{L} \mathbf{d}_s \mathbf{d}_s^T - L\mathbf{I} \tag{64}$$

where $\mathbf{I}$ represents the identity matrix.

Since $\mathbf{d}_s \in (-1,+1)^N$, then $||\mathbf{d}_s||_2^2 = N$. Using (64) and the orthogonality properties of the set $\{\mathbf{d}_s\}_{s=1}^L$ gives

$$[-\mathbf{A}+\mathbf{W}]\mathbf{d}_s = -(N-L)\mathbf{d}_s + (N-L)\mathbf{d}_s = \mathbf{0} \tag{65}$$

So, the prototype vectors $\{\mathbf{d}_j\}_{j=1}^L$ correspond to equilibrium points.

ACKNOWLEDGMENTS

This work was supported in part by Academy of Finland and Research Foundation (Tukisäätiö) of Helsinki University of Technology, Finland.

REFERENCES

[1] J.J. Hopfield and D.W Tank, Neural computation of decisions in optimization problems *Biological Cybernetics*, vol. :141-146, 1985.

[2] S. Matsuda, “Optimal” Hopfield network for combinatorial optimization with linear cost function, *IEEE Trans. Neural Networks*, vol. 9: 1319-1330, Nov. 1998.

[3] K. Smith, M. Palaniswami, and M. Krishnamoorthy, Neural techniques for combinatorial optimization with applications, *IEEE Trans. Neural Networks*, vol. 9: 1301-1318, Nov. 1998.

[4] K.C. Tan, T. Huajin and S.S. Ge, On parameter settings of Hopfield networks applied to traveling salesman problems, *Circuits and Systems I*, vol. 52, nr. 5: 994-1002, May 2005.

[5] J.K. Paik and A.K. Katsaggelos, Image restoration using a modified Hopfield network, *IEEE Trans. Image Processing*, vol. 1, nr. 1:49-63, Jan. 1992.

[6] T.S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice-Hall, New York, 1996.

[7] J.A. Farrel and A.N. Michel, A synthesis procedure for Hofield’s continuous-time associative memory, *IEEE Trans. Circuits Systems*, vol. 37: 877 - 884, 1990.

[8] G.G. Lendaris, K. Mathia and R. Saeks, Linear Hopfield networks and constrained optimization *IEEE Trans. Systems, Man, and Cybernetics, Part B*, vol. 29, nr. 1: 114 - 118 Feb. 1999.

[9] S. Haykin, *Neural Networks*, Macmillan, 1999.

[10] Z. Uykan, *Clustering-based algorithms for Radial Basis Function and Sigmoid Perceptron Networks*. PhD thesis, Control Eng. Lab., Helsinki University of Technology, 2001.

[11] Z. Uykan and H.N. Koivo, “A sigmoid basis nonlinear power control algorithm for mobile radio systems”, IEEE Trans. Vehic. Tech., 2003

[12] Z. Uykan, “From Sigmoid Power Control Algorithm to Hopfield-like Neural Networks: “SIR” (“Signal”-to-“Interference”-Ratio)- Balancing Sigmoid-Based Networks- Part I: Continuous Time”, *submitted to IEEE Trans. Neural Networks, 2008*.

[13] J.M. Zurada, *Introduction to Artificial Neural Systems*, West Publishing Company, 1992.

[14] M.K. Muezzinoglu and C. Guzelis, A Boolean Hebb rule for binary associative memory design, *IEEE Trans. Neural Networks*, vol. 15, nr. 1:195 - 202, Jan. 2004.

[15] M.K. Muezzinoglu, C. Guzelis and J.M. Zurada, An energy function-based design method for discrete hopfield associative memory with attractive fixed points *IEEE Trans. Neural Networks*, vol. 16, nr. 2:370-378, March 2005 .
1995.

[16] T. Huajin, K.C. Tan and Y. Zhang, A columnar competitive model for solving combinatorial optimization problems, *IEEE Trans. Neural Networks*, vol. 15, nr. 6: 1568 - 1574, Nov. 2004.

[17] J. Zander, “Performance of optimum transmitter power control in cellular radio systems,” *IEEE Trans. Veh. Technol.*, vol. VT-41, pp. 57-62, 1992.

[18] J. Zander, “Distributed cochannel interference control in cellular radio systems”, *IEEE Trans. Veh. Technol.*, vol. VT-41, pp. 305-311, 1992.

[19] D. O. Hebb , *The Organization of Behaviour* , John Wiley and Sons, New York, 1949.

[20] J. Van den Berg, “The most general framework of continuous Hopfield neural networks”, Proc. Int. Workshop on of Neural Networks for Identification, Control, Robotics, and Signal/Image Processing, pp. 92 - 100, 21-23 Aug. 1996.

[21] H. Harrer, J.A. Nossek and F. Zou, “A learning algorithm for time-discrete cellular neural networks”, Proc. IEEE Int. Joint Conf. on Neural Networks, vol.1, pp. 717 - 722, 1991.

[22] J.D. Herdtner and E.K.P. Chong, Analysis of a class of distributed asynchronous power control algorithms for cellular wireless systems, *IEEE Journal on Selected Areas in Comm.*, 18(3): 436 - 446, March 2000.

[23] Dongwoo Kim; On the convergence of fixed-step power control algorithms with binary feedback for mobile communication systems, *IEEE Transactions on Communications*, 49(2): 249 - 252, Feb 2001.

List of Figures

1 The figure shows percentage of correctly recovered desired patterns for all possible initial conditions in example 1 for the proposed D-Sgm”SIR”NN and Sign”SIR”NN as compared to traditional Hopfield network with 8 neurons. 121

2 The figure shows percentage of correctly recovered desired patterns for all possible initial conditions in example 2 for the proposed D-Sgm”SIR”NN and Sign”SIR”NN as compared to traditional Hopfield network with 16 neurons. 122

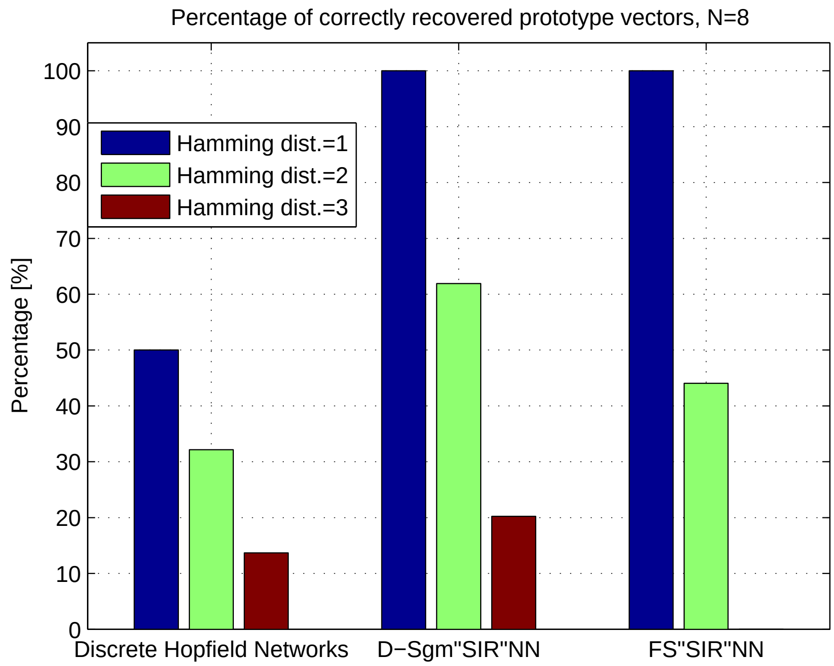
Percentage of correctly recovered prototype vectors, N=8
Percentage [%]
100
90
80
70
60
50
40
30
20
10
0
Hamming dist.=1
Hamming dist.=2
Hamming dist.=3
Discrete Hopfield Networks
D−Sgm"SIR"NN
FS"SIR"NN

Fig. 1.

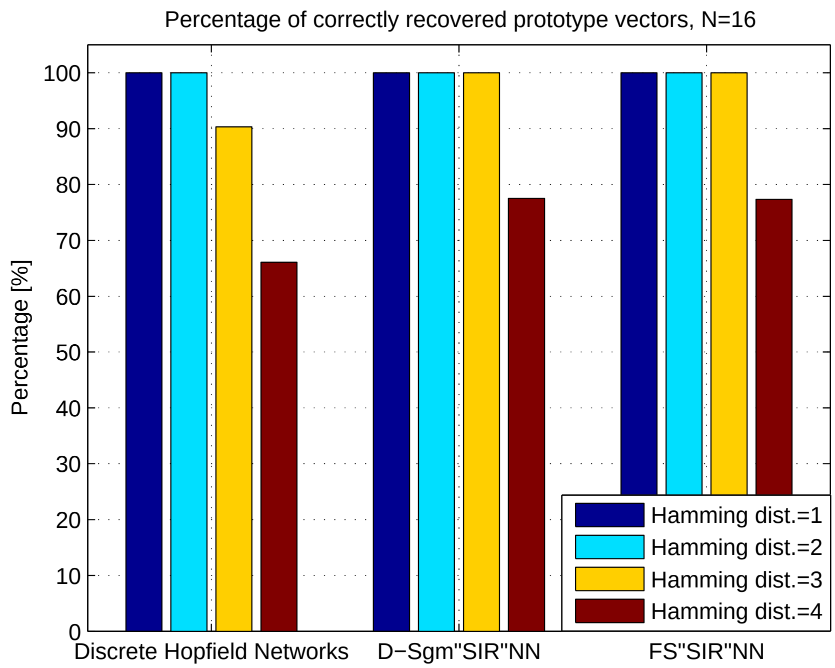
Percentage of correctly recovered prototype vectors, N=16
Percentage [%]
100
90
80
70
60
50
40
30
20
10
0
Discrete Hopfield Networks
D−Sgm"SIR"NN
FS"SIR"NN
Hamming dist.=1
Hamming dist.=2
Hamming dist.=3
Hamming dist.=4

Fig. 2.